

# Utcai objektumok gyors osztályozása LIDAR pontfelhősorozatokon

Börcs Attila<sup>1,2</sup>, Nagy Balázs<sup>1</sup> és Benedek Csaba<sup>1</sup>

<sup>1</sup> Számítógépes Érzékelés Kutatólaboratórium, Magyar Tudományos Akadémia, Számítástechnikai és Automatizálási Kutatóintézet

<sup>2</sup> Irányítástechnika és Informatika Tanszék, Budapesti Műszaki és Gazdaságtudományi Egyetem

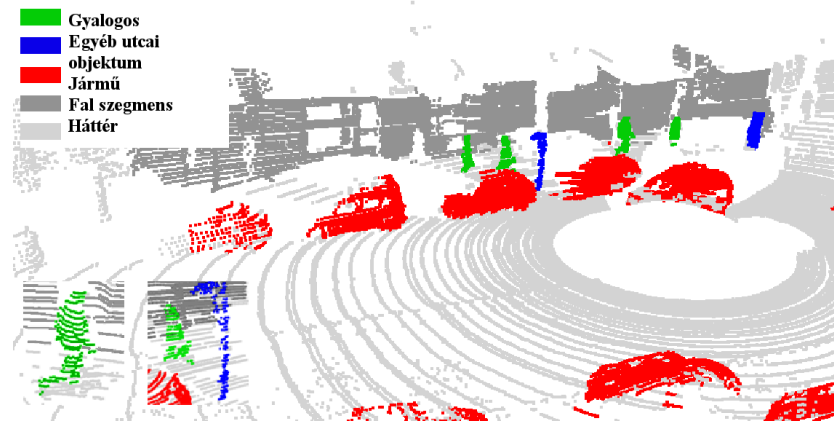
{vezetéknév.keresztnév}@sztaki.mta.hu

**Absztrakt.** Napjainkban a városi objektumok automatikus osztályozása egy kritikus jelentőséggel bíró feladat az önjáró robotok és járművek számítógépes érzékeléssel kapcsolatos feladatai közül. Cikkünkben egy új megoldást ajánlunk különböző városi objektumok osztályozására kültéri LIDAR pontfelhősorozatokon. A bemenetként használt háromdimenziós adathalmazt egy Velodyne HDL-64 típusú LIDAR lézerszkennerral készítettük különböző városi forgalmi szituációkban. Az általunk készített keretrendszer fogadja a szenzorból érkező pontfelhőfolyamot és célja, hogy lokalizálja és felismerje az adathalmazban elhelyezkedő összes járművet és gyalogost a mozgó szenzor hatósugarában. Első lépésként különböző régiókra szegmentáljuk a szenzorból érkező pontfelhőket a következő osztálycímkékkel: 1) talaj 2) alacsony objektumok 3) magas objektumok. Másodsorban, a kinyert 3-D előtérmaszkokon egy újszerű kétrétegű rács struktúra alapú területelárasztásos technikával objektum detekciót hajtunk végre, hogy pontosan becsüljük az egyes objektum kandidánsokhoz tartozó mérési pontokat. A szeparált 3-D objektumokból mélységképet készítünk, majd előzetesen - az objektum külső megjelenést figyelembe vevő - osztályozást hajtunk végre rajtuk egy konvolúciós neurális hálózat (Convolutional Neural Network - CNN) segítségével. Végül, a városi környezet topológiáját figyelembe véve, kontextuális jellemzőket vezetünk be, hogy finomítsuk (javítsuk) a kezdeti objektumosztályozást. A kifejlesztett eljárást egy valós városi környezetről készített, 1159 városi objektumot tartalmazó adathalmazon teszteltük, amelyek különböző forgalmi szituációkat foglaltak magukban.

## 1. Bevezetés

<sup>1</sup> A valós idejű háromdimenziós objektumfelismerés központi célkitűzése számos számítógépes látással kapcsolatos feladatnak, úgy mint automatizált közlekedés, önjáró járművek, vezetést segítő rendszerek [1, 2]. A nagysebességű lézerszkennerek, mint például a Velodyne HDL-64 LIDAR mérőrendszer nagyban képes támogatni ezen feladatok megvalósulását, hiszen segítségükkel lehetővé válik nagyméretű dinamikus utcai környezetek gyors háromdimenziós digitalizálása. A szenzor további nagy előnye,

<sup>1</sup> Benedek Csabát a Bolyai János kutatási ösztöndíj támogatta. A cikkben közölt eredmények jelenleg elbírálás alatt vannak egy nemzetközi szakfolyóiratban.



**1. ábra:** Az objektumosztályozás eredménye városi pontfelhőkön mélytanulás és kontextuális jellemzők használatával. Az osztályozott objektumok különböző színnel jelöltek.

hogy kompakt módon képes reprezentálni a környezetet 64000 pont/pillanatkép előállításával, feltételezve a valós idejű adatszolgáltatásnak és feldolgozásnak a mérőállomás és a fedélzeti számítógép között. Más mobil lézerszkennerekkel összehasonlítva (RIEGL VMX-250) [3, 4], a Velodyne pontfelhőkön történő objektumosztályozás kihívásokkal teli feladat több okból is. Elsősorban a berendezésből érkező adat zajjal terhelt, és számtalan olyan régió található a mért helyszínről, ahol a pontfelhő hiányos. Másodsorban, zsúfolt városi környezetben gyakran előfordul, hogy az egyes járművek, gyalogosok és egyéb utcai objektumok takarásba kerülnek. A takarásban lévő objektumok kinyert alakzatai gyakran hiányosak, vagy több darabra esnek a szét a pontfelhőben. Végül számolnunk kell a Velodyne LIDAR lézerszkennerek tipikus mérési karakterisztikájának kihívásaival, mint például az erősen csökkenő pontfelhőszűrűség a szenzortól távol eső régiókban.

A szakirodalomban számos módszer található, amelyek lézerszkennerekkel mért háromdimenziós adatokon kínálnak megoldást különböző felismerési feladatokra. Az [5] munkában a szerzők egy pontszintű objektum osztályozó eljárást mutattak be, bár a kiértékelés során csak egy Velodyne DHL-32E lézerszkennerek adatain végeztek kvalitatív összehasonlítást referencia (tanító) minták hiányában. A szerzők a [6] munkában egy objektum detekciós módszert mutattak be, ahol az osztályozás egyszerű téglatest modell oldal arányaira alapozva valósították meg. A [7] kifejlesztett módszer objektum és pontszintű jellemzőkből kiindulva végez *Support Vector Machine* alapú bináris osztályozást (*jármű/nem jármű*). Számtalan új módszer [8] használja kiértékelésre a LIDAR pontfelhőket is tartalmazó publikus adatbázist, a *The KITTI Vision Benchmark*-ot [1]. Az adatbázis fő limitációja, hogy a pontfelhőket érintő fél-automatikus címkézés csak egy előre néző optikai kamera látószögében történik, ami csupán a LIDAR pontfelhőknek egy csekély szegmensét érinti a 360° látószöget figyelembe véve. Megfelelő időfüggő modellek segítségével javíthatjuk az objektumfelismerés hatékonyságát [9], viszont zsúfolt utcai környezetben gyakran fordulnak elő forgalmi szituációk amikor

egy rendszernek azonnal kell döntést hoznia, és nincs lehetőség hosszú tér/idő dinamikák vizsgálatára. [10] egy jellemző-tanulás (feature learning) alapú módszer javasol utcai objektumfelismeréshez, továbbá 14 objektum kategóriában sorolva 588 objektumot tartalmazó tanító adatbázist publikál, bár ebben a munkában egy nyitott kérdés marad, hogy az objektum kinyerés minősége hogyan befolyásolja az objektumosztályozás pontosságát.

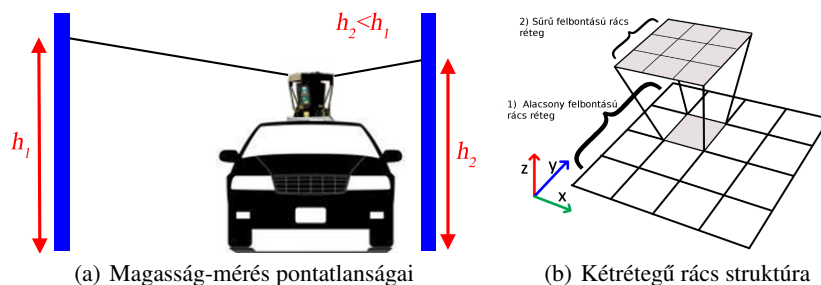
Ebben a munkában egy valós időben működő rendszert mutatunk be utcai objektumok kinyeréséhez és osztályozásához, ahol az eljárás objektum osztályozó modulja kifejezetten arra lett kifejlesztve, hogy hatékonyan dolgozza fel egy gyors objektum kinyerő eljárás [11] kimeneteit. Az algoritmikus modulok tesztelése lépésről lépésre történt. Kvalitatív és kvantitatív kiértékelés egy releváns referencia módszerrel összehasonlítva a cikk végén olvasható.

## 2. Az objektum osztályozó eljárás áttekintése

Ahogy az 1. ábra is szemlélteti, az elkészített rendszer célja, hogy felismerje és lokalizálja az összes járművet és gyalogost a LIDAR lézerszkennerek hatótávolságában. A folyamat négy egymás utáni lépésből áll. Először, a bemeneti pontfelhőn szegmentációt hajtunk végre, négy különböző osztálycímeként rendelve a pontokhoz: *talaj*, *alacsony előtér*, *magas előtér* és *ritka régiók*. Az *alacsony előtér* tipikusan utcai objektumokat (járművek, gyalogosok, padok, postaládák, hirdető táblák) tartalmaz, míg a *magas előtér* osztály homlokzatokat, házak falait, fákat, jelző lámpákat, póznáknak foglal magában. A szegmentációt követően a talajhoz és ritka régiókhoz tartozó pontokat eldobjuk, tekintve hogy a további feldolgozási lépésekben nem lesz szerepük. Másodsorban, a magas és alacsony előtér maszkokon objektum szeparációt hajtunk végre, így képesek vagyunk azonosítani, hogy mely 3-D mérési pontok tartoznak a lehetséges objektum kandidánsokhoz. Harmadszor, az *alacsony előtér* maszkból előzőleg kinyert objektumok előzetesen átmennek egy "külső megjelenés" alapú osztályozáson, ami segít megkülönböztetni a szintén lévő egyes járműveket és gyalogosokat az egyéb utcai objektumoktól. Ezzel párhuzamosan nagyméretű falszegmenseket - úgynevezett *horgony falakat* - detektálunk a *magas előtér*-ből kinyert objektum halmazból. Negyedszer, finomítjuk az előzőleg "külső megjelenés" alapján osztályozott objektumokat kontextuális jellemzők segítségével, figyelembe véve a különböző alacsony objektumok és a *horgony falak* relatív pozícióját.

### 2.1. Pontfelhő szegmentáció

A pontfelhő szegmentációt egy rács alapú megoldással valósítottuk meg [12] (2. ábra (b)). A rács struktúra kialakításához egy kétdimenziós  $S$  rácsot feszítünk a  $P_{z=0}$  síkra  $W_S$  rács cella mérettel, ahol  $s \in S$  jelöl egy önálló cellát a rács struktúrában. A talaj sík azonosításához a szenzor pozícióját használtuk referencia koordinátaként. A pontfelhő minden  $p \in \mathcal{P}$  pontját hozzárendeljük egy  $s_p$  cellához, ha az tartalmazza a  $p$  pont projekcióját a  $P_{z=0}$  talajsíkra. Jelölje  $\mathcal{P}_s = \{p \in \mathcal{P} : s = s_p\}$  azt a pontthalmazt ami az  $s$  cellába került levetítésre. Továbbá a cellákban eltávolítjuk a pontok magasság koordinátáit, és egyéb magasság jellemzőket, úgymint a maximális  $z_{\max}(s)$ , minimális



2. ábra: Falszegmensek magasságának a mérése, és a hierarchikus rács struktúra szemléltetése

$z_{\min}(s)$  és átlagos  $\hat{z}(s)$  magasságérték. Ezeket a jellemzőket később a szegmentációs és objektum szeparációs lépésben használjuk majd fel.

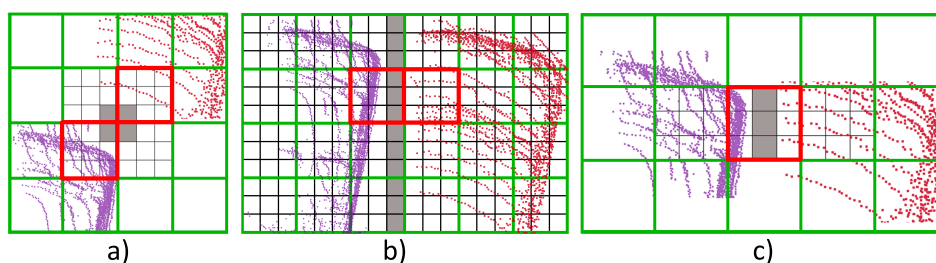
A szegmentálás során a cellákban lévő magasság információt használjuk annak érdekében, hogy osztálycímkéket tudjunk az egyes rács cellákhoz rendelni. Első lépésként megkeressük és eltávolítjuk azokban a cellákban lévő pontokat, amelyek darabszáma nem halad meg egy előre definiált küszöbértéket (tipikusan 4-8 pontot). Ezek a cellák gyakran zajos és ritkás - a szenzortól távol eső régiókban - találhatóak, és sok esetben megnehezítik és hátráltatják a felismerési feladatot. A ritka pontfelhő régiók eltávolítása után, a rács struktúra fennmaradó cellái *talaj* osztálycímkét kapnak, ha a minimális és maximális magasság érték egy cellán belül nem halad meg egy előre definiált küszöbértéket (mi 25cm-t használtunk munkánk során), továbbá az adott cella  $3 \times 3$  szomszédságában található cellákból származtatott átlagos magasságérték nem lép túl egy globális küszöbértéket. Egy cellához *magas előtér* osztálycímkét rendelünk, ha a maximális magasság egy cellán belül nagyobb mint egy előre meghatározott küszöbérték (140cm használtunk munkánk során), vagy a megfigyelt magasságkülönbség egy cellán belül nagyobb mint egy küszöbérték (310cm). Végül a fennmaradó - még osztálycímkét nem kapott - pontokhoz *alacsony előtér* címkét rendelünk.

A Velodyne LIDAR limitált vertikális látószöge miatt ( $+2^\circ$  felfelé,  $-24.8^\circ$  lefelé), a magasság alapú szegmentáció hibázhat a szenzor közelében. Ez a jelenség tipikusan keskeny utcákban tapasztalható, ahol az utca szélei túl közel helyezkednek el a mérőpozíciótól. Ebben az esetben az algoritmus téves osztálycímkéket rendel a szenzorhoz közeli rács cellákhoz. Ez a hiba ahhoz eredményez vezet, hogy az utcát szegélyező épületek homlokzataihoz vagy falaihoz gyakran *alacsony előtér* osztálycímkét rendel az algoritmus *magas előtér* helyett (2. ábra (a)). Definíció alapján, innentől ezeket a tévesen klasszifikált falszegmenseket *alacsony falszegmens*ként fogjuk hivatkozni, és a későbbi lépésekben felismerjük és szűrjük ezeket a problémás eseteket.

## 2.2. Objektum szeparáció

A kifejlesztett objektum szeparációs algoritmus három fő lépésből épül fel: *Először*, bejárjuk az alacsony felbontású rács réteg minden egyes celláját, és megvizsgáljuk minden  $s$  cella  $3 \times 3$  szomszédságát. A szomszédos cellák bejárása által lehetőségünk van egy cella lokális környezetéből származó jellemzőket számolni: (i)  $Z_{\max}(s)$  maximális

magasság érték az alacsony felbontású cellákon belül, és (ii) pontsűrűség (pontok darabszáma) a sűrű felbontású réteghez tartozó cellákból számolva. *Másodszor*, az algoritmus célja, hogy összetartozó pontok egy halmazát határozza meg a pontfelhőből szeparált előtér maszkon, úgy hogy az alacsony felbontású rács rétegen elhelyezkedő nagy cellákat összevonja (azonos osztály címkét rendel hozzá), abban az esetben ha a mért pontfelhőben lévő 3D-s pontok valóban közel helyezkednek el egymáshoz képest, és nagy eséllyel egy objektumhoz tartoznak. A  $\psi(s, s_r) = |Z_{max}(s) - Z_{max}(s_r)|$  kritérium segítségével azonos osztálycímke rendelhető azon cellákhoz az alacsony felbontású rács rétegen, ahol az  $s$  cella maximális magassága és a szomszédos  $s_r$  cellák maximális magasságai között mért különbség nem halad meg egy előre definiált magasság küszöböt. *Harmadszor*, végrehajtottunk egy finomítási lépést a detekciós eredményen a sűrű felbontású cellákat felhasználva. A magasság alapú kritérium gyakran nem működik megbízhatóan egymáshoz közel elhelyezkedő objektumok esetén, ugyanis az alacsony felbontású rács rétegen lévő cellák mérete túlságosan nagy ahhoz, hogy robusztusan kezelje ezeket az eseteket. A hibás detekciók kiküszöbölése érdekében megmérjük a cella kitöltöttséget a sűrű felbontású rács réteghez tartozó  $s'_d$  cellákban. Ahogy a 3. ábra is szemlélteti, azok a közel elhelyezkedő objektumok amelyek hibásan azonos osztálycímkét kaptak az alacsony felbontású cella rétegen, hatékonyan szétválaszthatók a sűrűbb rács rétegben elhelyezkedő cellákban történő pontsűrűség vizsgálatával. A következőkben bemutatunk néhány tipikus városi szituációt amikor az alacsony felbontású rács réteg hibásan egy objektumnak detektál két közel elhelyezkedő objektumot a pontfelhőben, viszont a sűrűbb rács rétegen ezek a hibás esetek kezelhetők. Ahogy a 3a) és 3b) ábrákon is látható, jelölje piros négyzet a két szomszédos cellát az alacsony felbontású rács rétegen. Mindkét esetben a nagy cellák egyaránt tartalmaznak pontokat az egyik, illetve a másik objektumból is, azonban a sűrű rács réteg kisebb celláin található olyan régiók (szürkével jelölve az 3a) - 3c) ábrákon) amelyek mentén elválasztható a két objektum. Ezekben a régiókban a pontsűrűség hirtelen változását vizsgáljuk az objektum szeparálás érdekében.

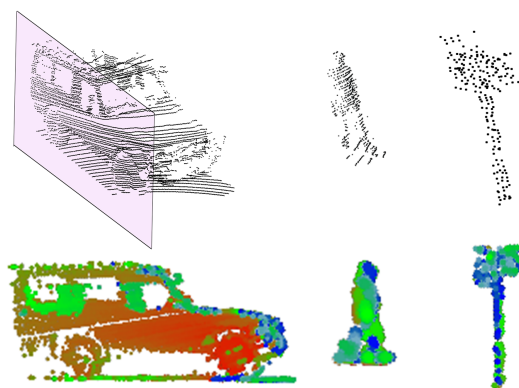


**3. ábra:** Közeleli objektumok szétválasztása a sűrű rács rétegen. [színek: zöld vonalak = alacsony felbontású rács réteg, fekete vonalak = sűrű felbontású rács réteg, szürke cellák: az objektum szeparáció során vizsgált régiók]

### 2.3. Objektumfelismerés külső megjelenés alapján

A következő lépés a feldolgozás során, hogy azonosítsuk a gyalogosokat és járműveket a 2.2. fejezetben előzőleg kinyert összefüggő pontfelhő szegmensek közül. Az előzetes feltételezésünk az, hogy ezek az objektumok nagy valószínűséggel az *alacsony előtér* régiókban fognak elhelyezkedni, ezért egy külső megjelenés alapú osztályozást hajtunk végre az előzőleg kinyert *alacsony* objektum várományosokon.

**Alacsony objektumok** Az osztályozó eljárás négy címkét vesz figyelembe: A *jármű* és *gyalogos* osztályokon kívül létrehoztunk egy *alacsony falszegmens* címkét, amely az *alacsony előtér*ben a falak magasságának limitált méréséből kifolyólag (2. ábra (a)). A fennmaradó *alacsony objektumokhoz* (padok, postaládák, bokrok, hirdető táblák) úgynevezett *egyéb utcai objektumok* osztálycímkét rendelünk. Az objektumok felismerését ellenőrzött tanulással valósítottuk meg: Először 2-D mélységképeket hozunk létre az objektum várományosokból, amely a továbbiakban egy gépi mélytanulás alapú eljárással osztályozunk. Az osztályozás kimenete minden bemeneti pontfelhőre vonatkoztatva négy konfidencia értéket tárol, amelyek az egyes osztálycímkékhez tartozó (*jármű*, *gyalogos*, *alacsony falszegmens*, *egyéb utcai objektumok*) osztályi valószínűségeket tartalmazzák. Annak érdekében, hogy hatékony jellemzőterképet hozunk létre az osztályozónak, egyenletesen mintavételezett mélységképeket hozunk létre (4. ábra) hasonlóan a [10] megoldáshoz.



4. ábra: Mélységkép előállítása kültéri LIDAR objektum várományosokból.

A hatékony objektumfelismerés érdekében jellemző tanulást (*feature learning*) alkalmaztunk egy konvolúciós neurális hálózat (*Convolutional Neural Network (CNN)*) segítségével. Munkánk során a *Theano* [13] keretrendszert használtuk. A CNN keretrendszer fogadja az előzőekben kinyert mélységképeket  $96 \times 96$  méretre skálázva. A keretrendszer kimenete négy  $[0,1]$  tartományban változó konfidencia érték, amelyekből következtethetünk, hogy az osztályozás során a CNN keretrendszer mennyire volt biztos az osztályozási döntésében a négy objektum kategóriát szem előtt tartva.

Különböző CNN architektúrákat tesztelve azt tapasztaltuk, hogy négy pár *convolution-pooling* rétegből csatolt *fully connected dense* réteg szolgáltatja a leghatékonyabb osztályozási eredményt.

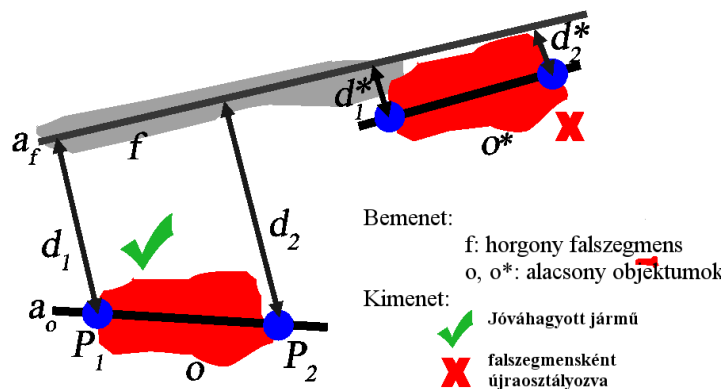
**Magas objektumok** Egy gépjárműre szerelhető Velodyne LIDAR konfiguráció nem a leghatékonyabb eszköz, ha magas utcai objektumok (jelző lámpák, hirdető oszlopok stb.) felismerését és lokalizálását akarjuk megvalósítani. A fő probléma, hogy a szenzor vertikális látószöge nem mindig elegendő az ilyen típusú utcai objektumok letapogatásához, ezért ezek az objektumok gyakran kiesnek a szenzor látómezőjéből. Ebben a cikkben nem fókuszálunk ezeknek az objektumoknak a megkülönböztetésével, de az előzőkben definiált *magas előtér* régiókból kinyerünk olyan nagyméretű falszegmenseket, amelyeket referencia pontként fogunk használni a 2.4. fejezetben részletezett kontextuális analízis kapcsán. Ezek a falszegmensek általában hosszúkas morfológiával bírnak, így detektálásukat egyszerűen a fő-irányukra vonatkoztatott méret küszöbölésével valósítottuk meg. Az így kinyert falszegmenseket, innentől *horgony falként* nevezzük, mivel képesek vagyunk segítségükkel becsülni az utcák határait (széleit).

## 2.4. Objektumfelismerés finomítása kontextuális analízissel

A 2.3. fejezetben bemutatott objektum klasszifikáció tisztán az objektumok külső megjelenése alapján kategorizál az objektumok mélységképein, viszont nem veszi figyelembe a szintéren elhelyezkedő egyéb tényezőket. Mivel zsúfolt utcai környezetben gyakran kell számítanunk teljes vagy részleges takarásokból adódó hiányos objektumokra, a tisztán megjelenés alapú osztályozás nem mindig fog eredményesen működni. Tapasztalataink alapján a CNN alapú osztályozó modul gyakran téveszti össze a *jármű* osztályú objektumokat a *falszegmensekkel*, mivel hasonló alaki tulajdonságokkal bírnak. Annak érdekében, hogy csökkentsük ezeket a hibás osztályozásokat, bevezetünk egy kontextuális jellemző alapú utófeldolgozó lépést, ahol kihasználjuk a topológiai kapcsolatokat a szintéren elhelyezkedő különböző objektumok között. Tipikusan a következő három szituációt kell lekezelnünk:

- A *járművekhez* és *egyéb utcai objektumokhoz* képest hasonló alaki tulajdonságokkal rendelkező objektumok gyakran megjelenhetnek tévesen a falszegmensek között. Ez a hiba javítható, ha megvizsgáljuk, hogy a 2.3. fejezetben bemutatott *alacsony objektumok* és a *horgony falszegmensek* fő iránya és távolsága milyen mértékben különbözik.
- Néhány, az út közepén elhelyezkedő objektumnak nagyon hasonló (általában magas) CNN konfidencia értéke van a *jármű*, illetve *alacsony falszegmens* osztálycímeként tekintve. Ez az eset tipikusan akkor fordul elő, ha két közel elhelyezkedő objektumot az objektum szeparáló modul (lásd 2.2. fejezet) tévesen egy objektumként kezel.
- A nagy oldalfelületekkel rendelkező *hosszúkas* járművek (pl.: kamion, teherautó, villamos) általában *falszegmenseként* lesznek detektálva, nagy CNN konfidencia különbséggel a *jármű* objektumosztályhoz viszonyítva. Tapasztalataink alapján a Velodyne pontfelhők alacsony felbontása miatt ezek az esetek nem választhatók szét

hatékonyan az objektumok külső megjelenését vizsgálva, így kénytelenek vagyunk a szintér topológiájára támaszkodni.

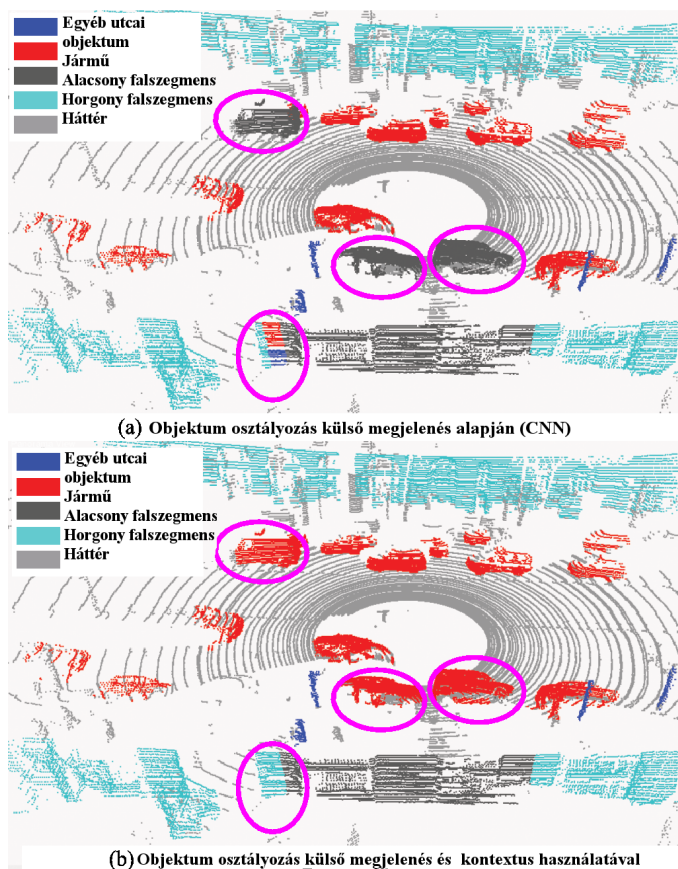


5. ábra: Kontextuális analízis: objektum - horgony irány és távolság kalkuláció

A kontextuális analízist végző modul fogadja a szintérről kinyert összes *horgony falszegmens* és *alacsony objektum* kandidánst. Az *alacsony objektumok*hoz már előzőekben hozzárendeltük az osztályokhoz tartozó négy konfidencia értéket (*jármű, gyalogos, falszegmens, egyéb utcai objektumok*) és a egy előzetes osztálycímkét, amihez a CNN modul a legmagasabb konfidencia érték rendelte. A külső megjelenést figyelembe vevő tulajdonságokon kívül, egy a szintér topológiáját is hasznosító jellemzőt az úgynevezett irányérzékeny távolságot (*alignment distance*) is alkalmaztunk a detektált *alacsony objektumok* és a *horgony falszegmensek* között. A  $d$  irányérzékeny távolság kiszámítását a 5. ábra szemlélteti; az ábrán a *horgony falszegmenst*  $f$ , míg a két *alacsony objektumot*  $o$  és  $o^*$  jelöli. Először vizsgáljuk meg a  $d(o, f)$  távolságot: főkomponens analízis segítségével megbecsüljük a demonstrációs ábrán látható két objektum ( $f$  és  $o$ ) fő-irányát ( $a_f$ ,  $a_o$ ). A továbbiakban meghatározunk az  $o$  objektum  $a_o$  tengelyén elhelyezkedő két végpontot  $P_1$ ,  $P_2$ , továbbá legyen  $d_1$ ,  $d_2$  két távolság az  $a_f$  objektum  $P_1$ ,  $P_2$  pontjai között. Az irányérzékeny távolságot az előzőekből a következő formula szolgáltatja:  $d(o, f) = \frac{d_1 + d_2}{2}$ ,  $d(o^*, f) = \frac{d_1^* + d_2^*}{2}$ . Ebben a példában szintér topológiát figyelembe vevő irányérzékeny távolság alapján  $o$  valódi jármű lehet, míg  $o^*$  valószínűleg egy falszegmens.

A detektált *alacsony objektumok* kontextus alapú újra-osztályozását a 1. algoritmus részletezi, ahol a  $IsConfident(o_i)$  függvény logikai igaz értékkel tér vissza, akkor és csakis akkor, ha az  $o_i$  objektumra vonatkoztatott első és második legnagyobb CNN konfidencia érték aránya nagyobb mint 0.8.





**6. ábra:** A tisztán külső megjelenés alapú (a) és kontextuális jellemzőkkel kibővített osztályozás (b) összehasonlítása

**Input:** Előzetesen osztályozott *alacsony objektumok* halmaza  $O = \{o_1, o_2, \dots, o_n\}$   
**Input:** *Horgony falszegmensek* halmaza  $F = \{f_1, f_2, \dots, f_m\}$   
**Output:** Kontextus analízis után módosított objektumok halmaza  $O = \{o_1, o_2, \dots, o_n\}$

```

for  $i \leftarrow 1$  to  $n$  do
    if  $\min_{f \in F} d(o_i, f) < \mu$  then
        Label( $o_i$ )  $\leftarrow$  Facade
    else if Label( $o_i$ ) = Facade then
        if !IsConfident( $o_i$ ) then
            Label( $o_i$ )  $\leftarrow$  Vehicle
        else
            Label( $o_i$ )  $\leftarrow$  LongVehicle
    end
end
return  $O = \{o_1, o_2, \dots, o_n\}$ 
    
```

**Algorithm 1:** Objektumok osztálycímke módosítása kontextuális információ felhasználásával.

### 3. Kiértékelés

Mivel az általunk kifejlesztett eljárás egy teljes folyamatot mutat be: a pontfelhő szegmentálástól, az objektum szeparáción át egészen a felismerésig; a publikus pontfelhő adatbázisok (Sydney [10], Stanford [9]) magukban nem megfelelőek az eljárásunk validálására. Egy másik opció a kiértékelésre a szintén publikusan elérhető *KITTI benchmark*, bár az adathalmaz nem tartalmaz annotált falszegmenseket, amelyek detektálása az általunk kifejlesztett keretrendszer szerves részét képezi, ezért a kiértékelést egy saját kézzel annotált adatbázison végeztük (SZTAKI *Velodyne64Road*). Az adatbázis egy gépjárműre erősített Velodyne HDL-64 LIDAR szenzor által készített pontfelhő szekvenciákat tartalmaz; a felvételek Budapest belvárosában készültek. Az adatbázis a következőképpen készítettük: a begyűjtött teszt adathalmazon először lefuttattuk a 2.1. és 2.2. fejezetekben már részletezett pontfelhő szegmentáló és objektum kinyerő algoritmust. További lépésként kézzel osztályoztuk az automatikusan kinyert 2063 darab alacsony objektumot *jármű*, *gyalogos*, *alacsony falszegmens* és *egyéb utcai objektumok* kategóriákban, így a vizsgált objektumokat képesek vagyunk egy pontfelhőszegmensen reprezentálni, amely egy valós körülmények között működő objektum kinyerő algoritmus kimenete, ami magában foglalja az összes takarást, hibás vagy részlegesen kinyert objektumokat stb.

A kiértékelés során a kifejlesztett eljárás utolsó két lépésére koncentráltunk: a tisztán külső megjelenést figyelembe vevő CNN alapú objektumosztályozásra (2.3. fejezet), és kontextuális jellemzők általi finomított objektum klasszifikációra (2.4. fejezet). A CNN osztályozó betanításához elválasztottuk a saját adathalmazunkból 904 objektumot. A tanító adatbázis összesen 402 *járművet*, 261 *alacsony falszegmenst*, 467 *egyéb utcai objektumot* és 208 *gyalogost* tartalmazott. A tesztelésre használt adatbázis - a tanításhoz nem használt - maradék 1159 objektumokból épül fel 588 *járművel*, 72 *alacsony falszegmennel*, 452 *egyéb utcai objektummal* és 77 *gyalogossal*. A kiértékelés során megszámoltuk a helyesen és helytelenül osztályozott objektumokat, és kiszámoltuk a fedést, pontosságot és F-mértéket minden egyes objektumosztályra egyenként, illetve összesítve is. A tisztán külső megjelenés alapú illetve kontextuális jellemzők segítségével finomított detekciós eredmények kvantitatív kiértékelését a 1. táblázat foglalja össze. Szignifikáns javulás figyelhető meg az osztályozás pontosságát illetően a *jármű* és *alacsony falszegmens* objektumosztályokat figyelembe véve, különös tekintettel a hamis riasztások (*false alarm*) tekintetében. A cikkben ajánlott módszer egy friss szakirodalmi eljárással is összehasonlításra került [14]. A szerzők által kifejlesztett módszert eredetileg nagy-felbontású mobil lézerszkennerekkel tesztelték, bár az eljárást - a sokkal ritkább felbontású - Velodyne pontfelhőkre is alkalmazni lehet, viszont a teljesítménye nem haladja meg az általunk ajánlott eljárását.

### 4. Összefoglalás

Ebben a munkában egy teljes algoritmikus és szoftveres rendszert ajánlottunk utcai objektumok osztályozásához Velodyne LIDAR pontfelhő szekvenciákon. Az eredmények igazolják, hogy az ajánlott objektum osztályozó eljárás robusztusan képes klasszifikálni egy kültéri lézerszenzor által mért háromdimenziós utcai objektumokat négy előre meghatározott objektum kategóriában.

**1. táblázat:** Az objektum klasszifikáció kiértékelése

OK	OSZ	Referencia módszer [14]			Saját megoldás, csak CNN			Saját megoldás, CNN és kontextus		
		Pr	Rc	Fr	Pr	Rc	Fr	Pr	Rc	Fr
<b>J</b>	558	90	95	93	<b>92</b>	<b>98</b>	<b>95</b>	<b>99</b>	<b>99</b>	<b>99</b>
<b>AFSZ</b>	72	79	48	59	<b>85</b>	<b>48</b>	<b>61</b>	<b>94</b>	<b>71</b>	<b>81</b>
EUO	452	88	95	93	92	97	95	92	97	95
Gy	77	69	56	62	79	77	78	79	77	78
Sum	1159	81	73	76	92	80	85	91	86	89

Jelölések: Objektum Kategória (OK): Jármű (J), Alacsony falszegmens (AFSZ), Egyéb utcai objektum (EUO), Gyalogos (Gy), Objektumok száma (OSZ), Precision (Pr), Recall (Rc), F-rate (Fr), %-ban kifejezve

## 5. Köszönetnyilvánítás

A szerzők szeretnék megköszönni Kovács Leventének és Varga Domonkosnak a segítséget és tanácsokat a mélytanulási eljárás implementálását és alkalmazását illetően.

## Irodalom

- Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2012)
- Börcs, A., Nagy, B., Benedek, C.: Dynamic environment perception and 4D reconstruction using a mobile rotating multi-beam Lidar sensor. In: Handling Uncertainty and Networked Structure in Robot Control. Studies in Systems, Decision and Control. Springer (2016) 153–180
- Yu, Y., Li, J., Yu, J., Guan, H., Wang, C.: Pairwise three-dimensional shape context for partial object matching and retrieval on mobile laser scanning data. IEEE Geoscience and Remote Sensing Letters **11** (2014) 1019–1023
- Guan, H., Yu, Y., Li, J., Liu, P.: Pole-like road object detection in mobile lidar data via supervoxel and bag-of-contextual-visual-words representation. IEEE Geoscience and Remote Sensing Letters **13** (2016) 520–524
- Maligo, A., Lacroix, S.: Classification of Outdoor 3D Lidar Data Based on Unsupervised Gaussian Mixture Models. In: IEEE International Symposium on Safety, Security, and Rescue Robotics, West Lafayette, USA0 (2015)
- Azim, A., Aycard, O.: Detection, classification and tracking of moving objects in a 3D environment. In: IEEE Intelligent Vehicles Symposium (IV), Alcalá de Henares, Spain (2012) 802–807
- Himmelsbach, M., Müller, A., Luettel, T., Wuensche, H.J.: LIDAR-based 3D Object Perception. In: International Workshop on Cognition for Technical Systems, Munich, Germany (2008)
- Wang, D.Z., Posner, I.: Voting for voting in online point cloud object detection. In: Proceedings of Robotics: Science and Systems, Rome, Italy (2015)
- Teichman, A., Levinson, J., Thrun, S.: Towards 3D object recognition via classification of arbitrary object tracks. In: International Conference on Robotics and Automation. (2011)

10. De Deuge, M., Quadros, A., Hung, C., Douillard, B.: Unsupervised feature learning for outdoor 3D scans. *Proceedings of Australasian Conference on Robotics and Automation* (2013)
11. Börcs, A., Nagy, B., Benedek, C.: Fast 3-D urban object detection on streaming point clouds. In: *Workshop on Computer Vision for Road Scene Understanding and Autonomous Driving at ECCV 2014*. Volume 8926 of *Lecture Notes in Computer Science*. Springer, Zürich, Switzerland (2015) 628–639
12. Józsa, O., Börcs, A., Benedek, C.: Towards 4D virtual city reconstruction from Lidar point cloud sequences. In: *ISPRS Workshop on 3D Virtual City Modeling*. Volume II-3/W1 of *ISPRS Annals Photogram. Rem. Sens. and Spat. Inf. Sci.*, Regina, Canada (2013) 15–20
13. Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I.J., Bergeron, A., Bouchard, N., Bengio, Y.: Theano: new features and speed improvements. In: *Deep Learning and Unsupervised Feature Learning Workshop at NIPS, Lake Tahoe, USA* (2012)
14. Rusu, R.B., Cousins, S.: 3D is here: Point cloud library (PCL). In: *International Conference on Robotics and Automation, Shanghai, China* (2011)